基于嵌入式 AI 实现的智能投放无人机

夏驰，王瑞祺，陈锦轩

第一部分　设计概述

1.1 设计目的

游泳池时常发生溺水事件，因此游泳池会有专业安全员来保证人们游泳时的安全。此外，水库,野河道等禁游区虽然明文规定禁止人员游泳,但每年仍有很多人违反规定在该类地点游泳. 导致溺水事故时常发生。为避免相关事故的发生,禁游区域的管理人员常采用护栏隔离或者雇佣巡逻人员的方法。这些方法对于小范围区域有一定效果，但对于大范围水域，效率低下且成本较高，因此亟需**高效，低成本**的方法以替代。[1]

该系统主要有三个目的:

1: 利用计算机视觉对人进行检测,并通过相关算法保证检测结果的准确性。

2: 利用无人机按照设定的路径进行自动巡查,使可巡查的范围更广且巡查效率更高, 降低巡查成本。[2]

3: 为水中人员进行救援物品投放, 提高救援成功率。

1.2 应用领域

该作品可应用于水域巡查及救援物资的投放。该作品可显著提高被救援者的成功救援率，且对于情况未知的危险水域，该作品保障了救援人员救援过程中的自身安全。该作品也可替代安全员对泳池进行监管。

该作品针对于以下场景.

1: **水库等禁游区**的无人巡查(或是有人的辅助巡查), 巡查时可携带救生圈, 当检测到被救援人员时向其附近投出救生圈

2: 已知目标区域有人在水中处于危险,派出无人机向目标区域,当检测到目标时,扔出救生设备。

3: 对**泳池**进行监管，可起到替代安全员的作用，为泳池中的溺水人员投放救援物资。[3]

1.3 主要技术特点

(1) **嵌入式 AI：**

基于 darknet 架构，改良后的 yolo-fastest 算法，VOC2007/2012 数据集训练神经网络，并通过模型量化最终得到 tflite 类型文件，使用 RT-Thread 团队为 RT-Thread 实时操作系统所开发的 AI 套件 RT-AK 将 AI 植入到 Art-Pi 上，实现 AI 模型在边缘设备上的运行。

(2) **无人机**:

使用无人机为运动载体, 高适应性，灵活性使其可适用于多种不同的地形。同时，无人机的飞控集成了加速度计，陀螺仪，气压计等传感器，可测定无人机飞行中的各项数据。

(3) 机载电脑

机载电脑可与 PC 端远程连接，通过 PC 可显示无人机的飞行时状态数据，例如高度，速度，实现远程监控。

(4) 板间通讯

无人机各模块间通过不同通讯协议如进行通讯，使本来独立的模块形成完整的系统。

(5) 智能投放

该无人机除了螺旋桨电机外还搭载了抛物器，一次飞行可携带多个救援物品，进行两次投放。

1.4 关键性能指标

1. 续航能力：续航能力会直接决定无人机的救援效果，良好的续航能力将提高无人机的可靠性。在速度一定的情况下，无人机的续航能力主要却决于两个方面：电池与负载，在带有一定负重的条件下，电池越好，续航能力越强。我们无人机一次充电总巡航时间要达到 15 分钟以上。

2. 飞行高度/速度: 飞行高度及速度都会影响识别率，且高度及速度都应位于安全范围内。 高度过高，摄像头与人距离过远，难以识别，同时无人机的飞行速度受限于 Art-Pi 进行图像处理的速率。在特定飞行高度和速度下，无人机的准确识别率应该达到最高。

3. 巡航能力: 巡航能力指无人机实际飞行过程中是否会按照设定的路径飞行，并且最终是否能够回到起始点。巡航能力主要受电池与飞行速度所影响。在实验过程中可通过测距并设定标志物来判断无人机是否正常巡航。

1.5 主要创新点

(1) AI 模型优化及轻量化

边缘设备资源有限，大型神经网络不适用于边缘设备。进行神经网络训练时，为了使最终得到的 AI 模型能够运行在 Art-Pi 上， 我们去掉了特征金字塔输出，只保留一个输出，同时删减网络结构，将网络层数由 100+删除到 20+，保证对人体的检测有效，经过后续将 yolo-fastest 模型转换成 keras，最终转换成 tflite，成功地植入 Art-Pi 进行图像推理。

(2) 通过机载电脑实现无人机控制

传统的无人机需要通过遥控器控制，使用人员需要进行相关训练后才能熟练地操作无人机。我们利用 MAVLink 协议实现了机载电脑与飞控间的通讯，通过机载电脑间接控制无人机的飞行。通过编写程序，规划路线，运行程序，无人机便能按照程序所设定的数据自动巡航，无需额外操作，降低了人员的培训成本。

(3) 智能投放

无人机上安装了抛物器，可进行两档开合，进行两次投放。机载电脑树莓派相连，通过 PWM 信号控制抛物器开合，调制脉宽周期可实现抛物器的两档开合。我们通过 UART 通讯协议实现了机载电脑与人体识别模块核心板 Art-Pi 之间的串口通讯，当人体识别模块识别到人体时，将向机载电脑传数据，机载电脑接收到数据并控制抛物器开合。

第二部分　系统组成及功能说明

2.1 整体介绍

2.1.1 硬件部分

整个智能投放无人机由**人体识别模块，机载电脑，飞控，电机，供电模块**共五个部分组成。

作品硬件设计框架如图 1：

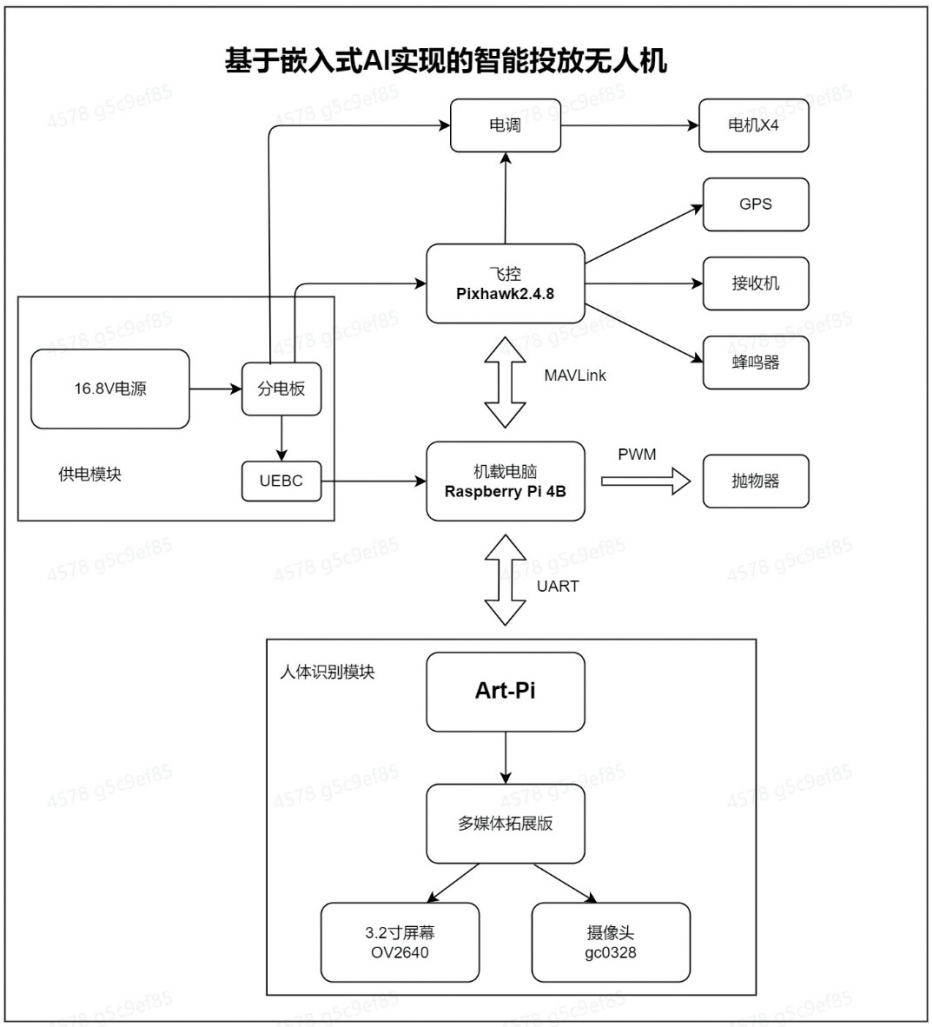


图 1 无人机硬件系统整体架构

如图 2-图 5 所示，无人机从上到下可分为 4 层，飞控硬件 pixhawk2.4.8 位于第一层，用以控制无人机的飞行状态。分电板与电调位于第二层，用以系统供电。人体识别模块位于第三层，可实现人体识别。机载电脑和电池位于最底层，机载电脑与 PC 端连接实现无人机远程控制，电池是系统总电源。

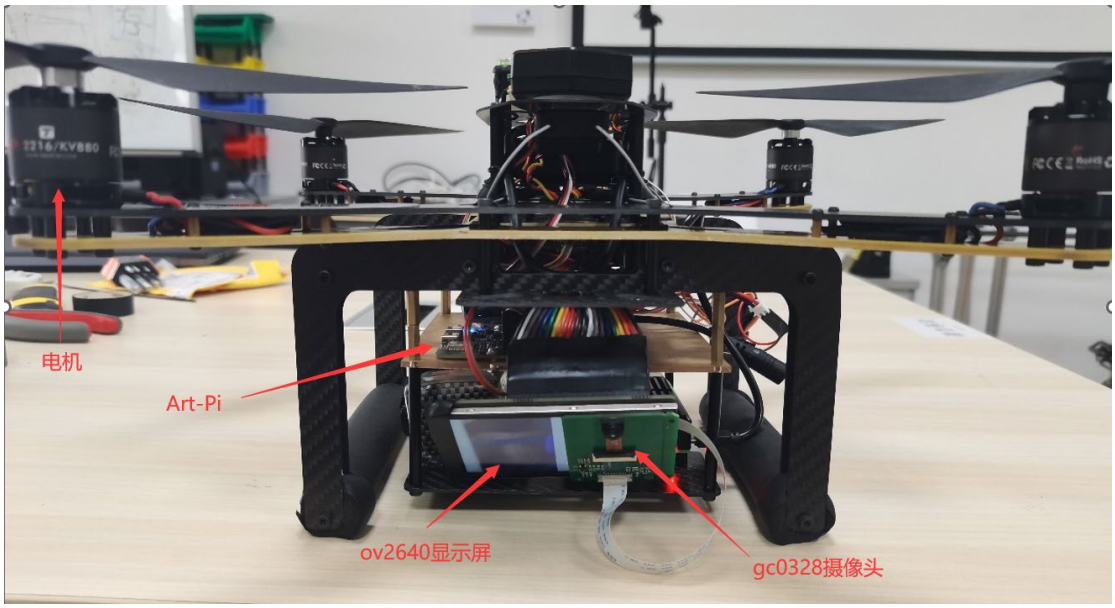

电机
Art-Pi
ov2640显示屏
gc0328摄像头

图 2 主视图

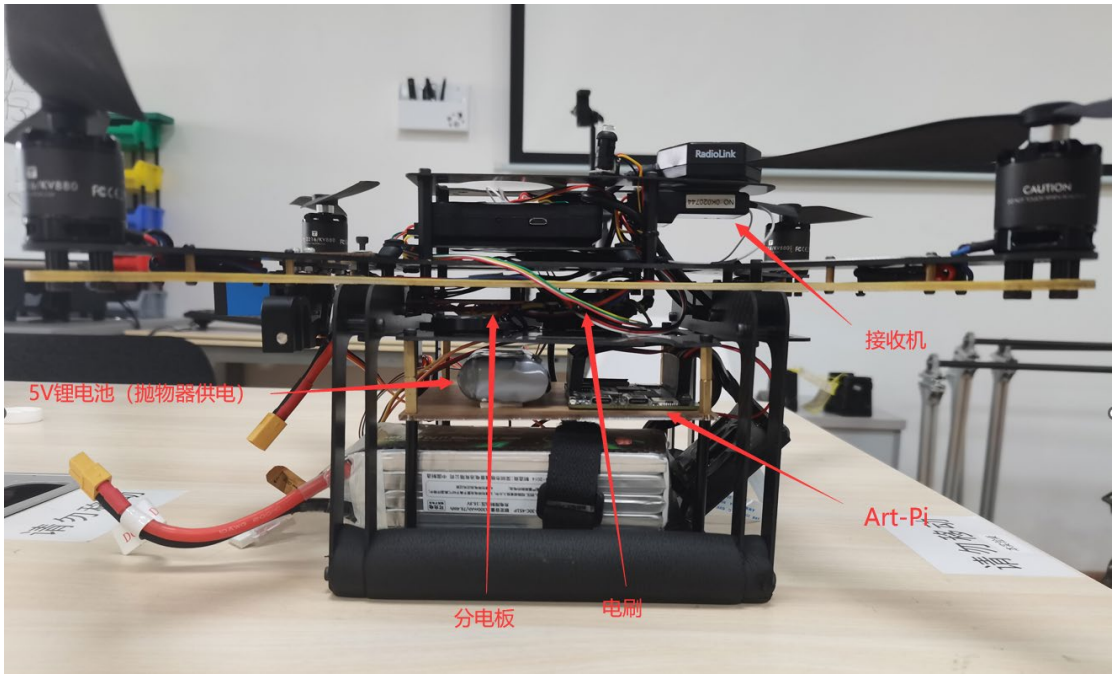

RadioLink
CAUTION
接收机
5V锂电池（抛物器供电）
Art-Pi
分电板
电刷

图 3 左视图

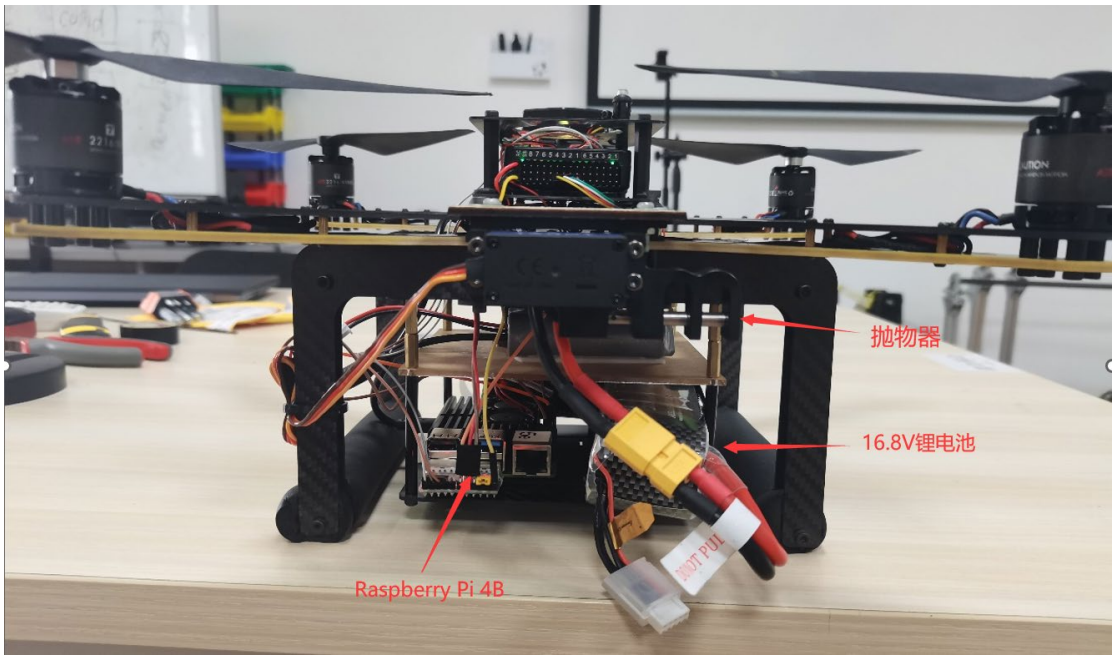


图 4 后视图

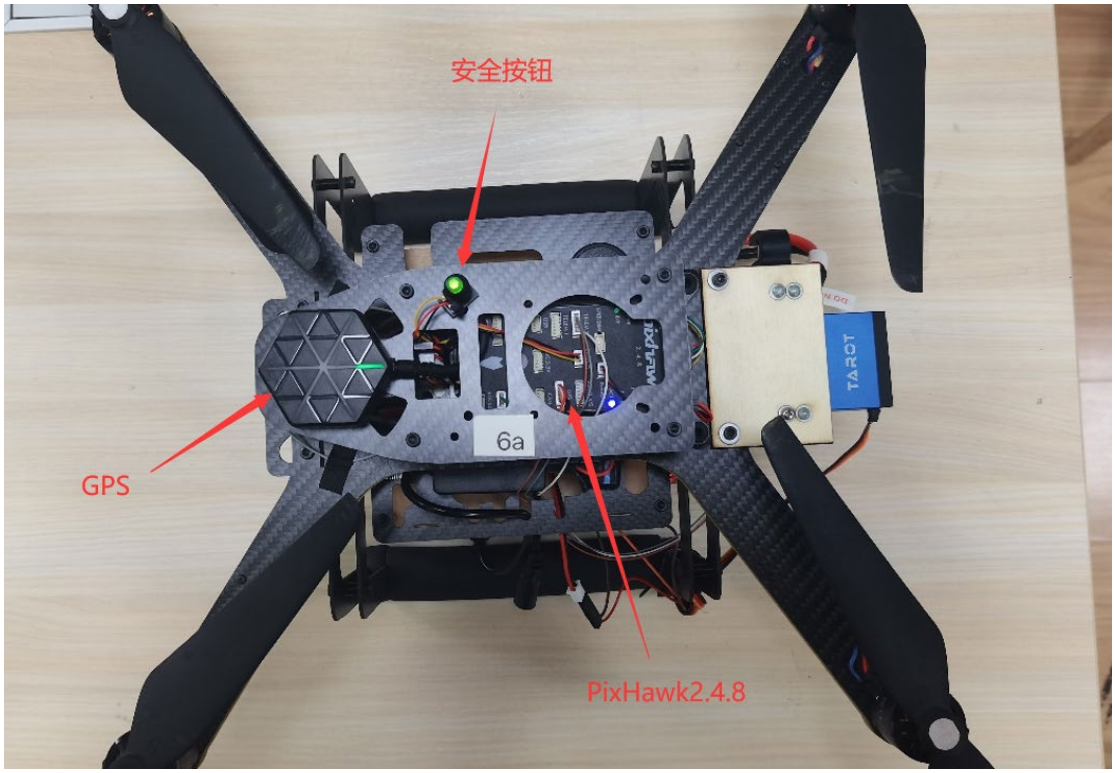


图 5 俯视图

2.1.2 软件部分

我们的作品软件系统主要分为三个部分，分别在飞控 Pixhawk，机载电脑 Raspberry Pi 4B，Art-Pi 上运行。

软件设计架构如图 6 所示：

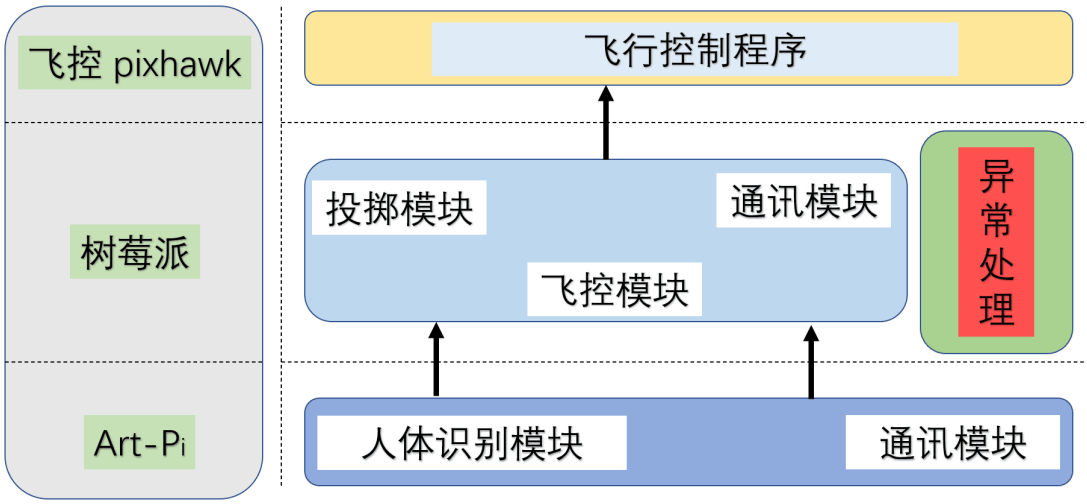


图 6 软件设计架构

工作流程图如图 7 所示：

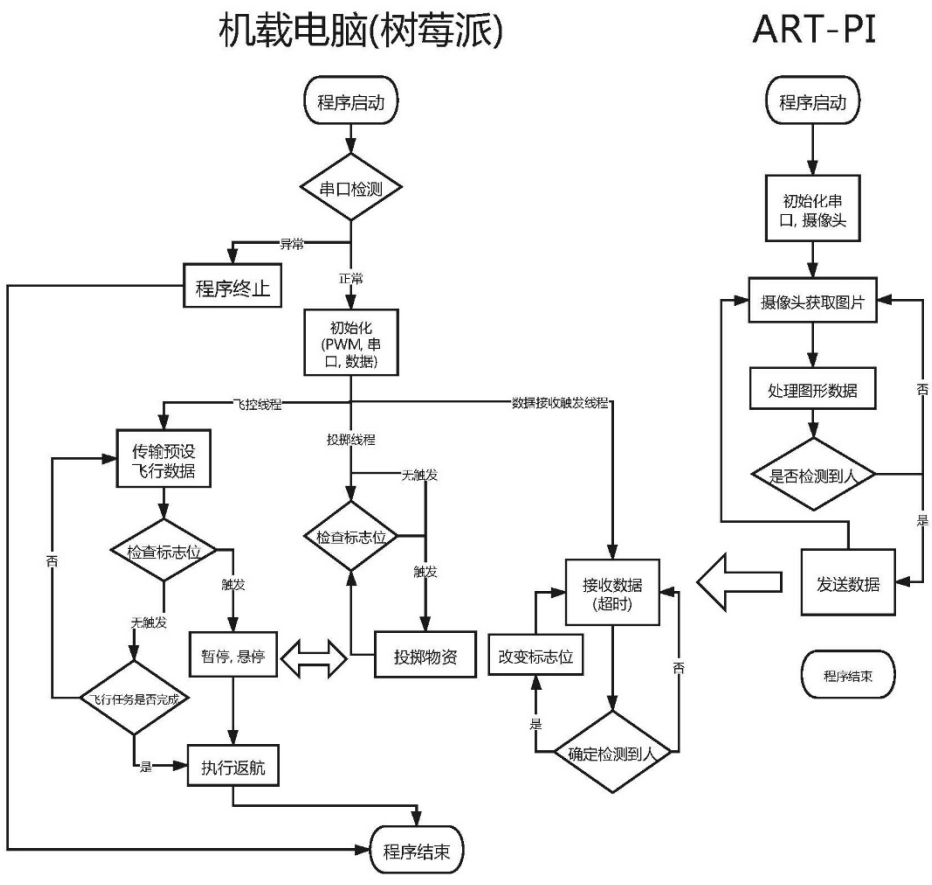


图 7 软件流程图

Art-Pi 端：程序启动后，先进行串口和摄像头初始化以保证后续程序正常执行，从摄像头获取每帧画面经过图片预处理后输入部署上的神经网络进行推理，得到推理结果后通过串口将结果传给树莓派。

树莓派端:程序启动后将进行串口检测保证通讯的正常,当检测成功后各项数据和硬件都将被初始化,此后程序将分为**三个线程**进行对应的工作.

1. 飞控线程：负责控制无人机飞行并轮询检测投掷标志位,当标志位触发时将控制无人机进行悬停,之后无人机将进行返航;
2. 抛投线程：用于轮询投掷标志位,当标志位触发就会进行物资地抛投;
3. 接收线程：用于超时接收 ART-PI 传回的数据并对其进行判断,当判断到有效目标后该线程会改变标志位用于触发其他两个线程的相关工作.

2.2 各模块介绍

2.2.1 硬件模块介绍

(1) 人体识别模块:

人体识别模块核心板为 **Art-Pi**, 在无人机工作过程中，人体识别模块充当了“眼睛”和“大脑”的角色，扫描救援路径并检测是否有需要救援的人员。**Art-Pi** 上部署了我们训练的**神经网络模型**, 通过将其连接多媒体拓展板和摄像头, 编写对应的应用层代码，可进行图像的传输及对摄像头视野内的人体进行检测返回检测标志位。该模块包括 **Art-Pi**, 多媒体拓展版，摄像头。

1. **Art-Pi**: 运行人体识别程序，与多媒体拓展版通过杜邦线相连，与树莓派通过 USB 转 ttl 串口下载线连接。多媒体拓展板将图像数据传输给 **Art-Pi**, Art-pi 上运行应用层代码进行人体识别，通过 **UART 串口**与树莓派通讯，传输识别标志位。

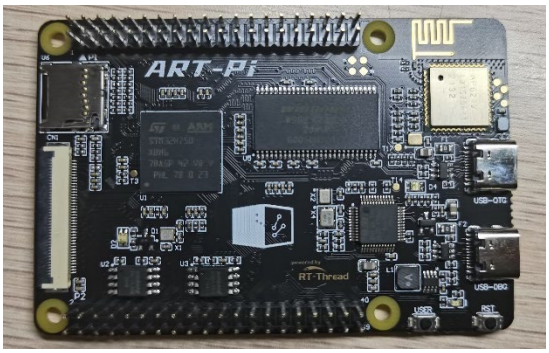


图 8 Art-Pi 实物图

2. 多媒体拓展版和摄像头：显示屏 ov2640, 摄像头 gc0328, 摄像头与拓展版通过排线连接，摄像头拍摄视频，显示屏上显示，显示屏主要用于开发过程中的调试，可直观地显示人体是否识别成功。

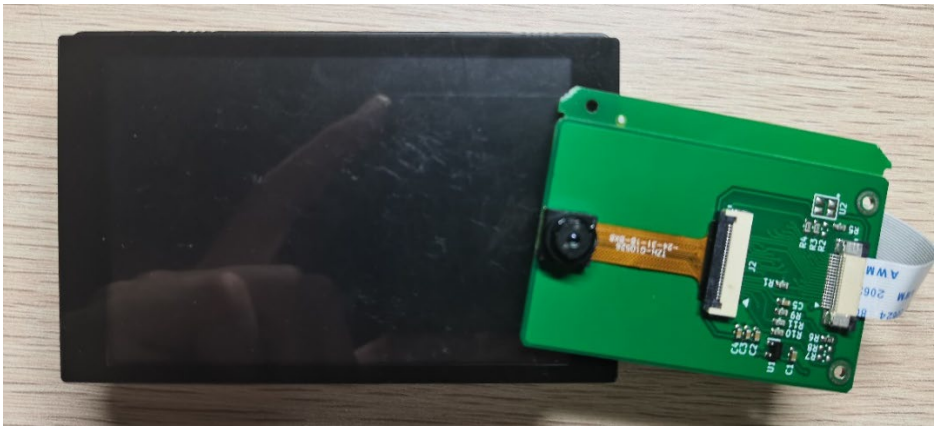


图 9 多媒体拓展版正面

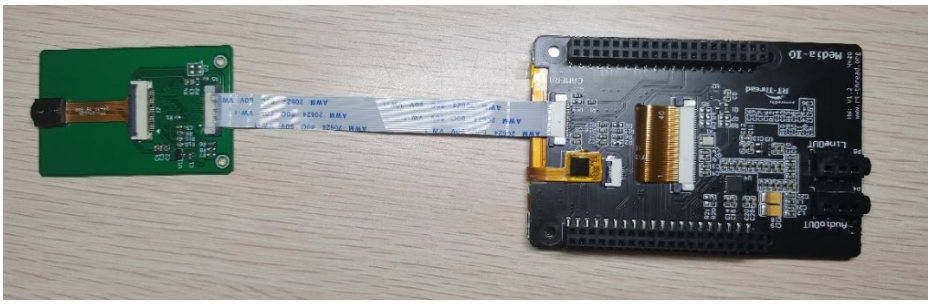


图 10 多媒体拓展版背面

(2) 机载电脑

我们采用树莓派4B作为机载电脑，可运行外部控制程序，给飞控发送外部控制指令。我们为树莓派4B安装了ubuntu20.04系统，可以运行基于dronekit开发的python飞控脚本。给树莓派设定开机自建热点，PC端可直接连接树莓派的WIFI热点组建局域网，通过SSH命令远程连接树莓派，远程终端运行飞控脚本即可控制无人机，并且可读取到目前飞控的数据：系统警报、固件版本、姿态数据、电池电量、解锁状态、当前飞行模式等等。基于UART协议，我们实现了树莓派与Art-Pi之间的通讯，并且用树莓派引脚输出PWM信号控制抛物器的开合。根据Art-Pi的识别情况，树莓派就可控制抛物投放救援物品。

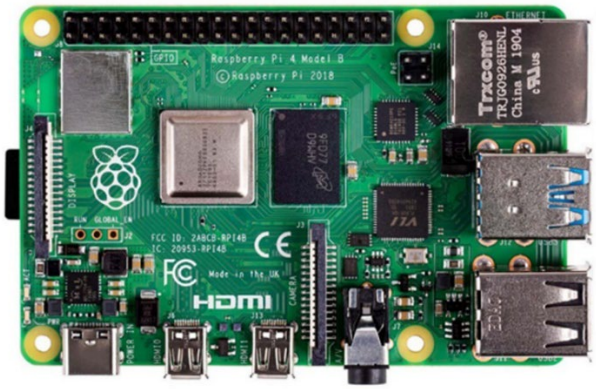


图 11 树莓派 4B

(3)飞控

我们飞控模块采用pixhawk2.4.8。Pixhawk2.4.8是一套集成了传感器、输入输出功能及嵌入式处理芯片的电路板，是无人机正常飞行的必要硬件，它是通过一系列的传感器感知无人机的高度、速度、位置等信息，在通过计算，对电调发出信号，从而间接控制电机转速，达到改变无人机姿态的目的。在我们无人机系统中，pixhawk2.4.8集成了ArdupPilot软件套件，并且可通过MAVLink通讯协议与树莓派4B通讯，运行树莓派上的脚本文件后，可通过pixhawk飞控间接地控制无人机的飞行姿态。



图 12　pixhawk2.4.8

(4) **电机**

该无人机系统的机械部分主要包括螺旋桨电机(T-motor2216), Tarot 抛物器

1. 螺旋桨电机：T-motor2216 受经由电调处理的 pixhawk 信号控制，电机旋转带动桨叶，从而给无人机提供上升的动力。
2. 抛物器：接收来自树莓派的 PWM 信号控制，控制机械爪的开合，从而投放物品。

我们团队最初曾采用中合金机械爪(如图 15 所示)来挂载无人机上的救援物资，但是经过后续实验发现，使用机械爪局限很多：

- 首先，机械爪运输能力有限，且只能进行一次投放，使无人机救援能力大大降低；
- 其次，机械爪自身重量较重，占据了无人机上本就宝贵的负载空间。

基于以上的发现，我们采用了负载能力更强，重量更轻的双抛抛物器(如图 16 所示)，提高了无人机救援能力。



图 13 T-motor2216



图 14 中合金机械爪



图 15 双抛抛物器

(5) 供电模块

该无人机系统的总电源为格式 4s 5300mah 锂电池，此外为了给无人机系统中的各个部分供电,我们还需要电调，分电板，UBEC

1. 电调：接收飞控的输出信号，处理后驱动电机旋转；
2. 分电板：将总电源进行分压，提供给各个部分；
3. UBEC: 可调输入电压，可设定 5-8V 内电压，为树莓派提供恒定电压。



图 16 格式 4s 5300mAh 锂铍电池

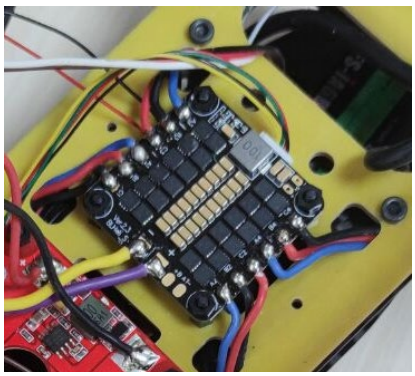


图 17 电调

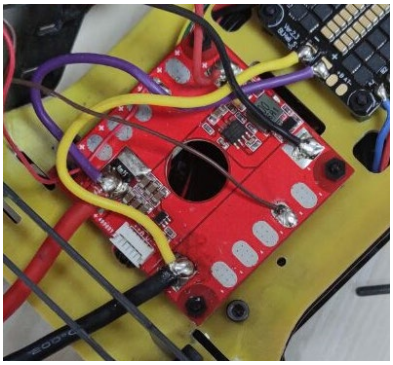


图 18 分电板

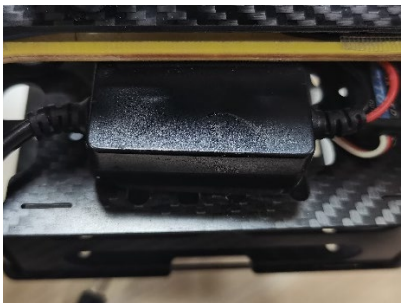


图 19 UBEC

2.2.2 软件模块介绍

(1). 飞控

我们选用 Pixhawk2.4.8 作为飞控硬件后，我们还需要飞控系统，也就是飞控固件来控制无人机的飞行状态。**Ardupilot** 是可信的，开源的飞控系统，可用于我们所使用的四轴无人机的飞行控制。**MAVLink 协议**是在串口通讯基础上的一种更高层的开源通讯协议，主要应用在微型飞行器的通讯上。我们在 **pixhawk2.4.8** 上安装了 **Ardupilot4.1.0** 的环境，并基于 **MAVLink** 协议实现了飞控与机载电脑间的通讯。

(2). 树莓派

选用树莓派作为主要通讯控制模块,通过在其上运行编写好的脚本以达到与人体识别模块和飞控模块通讯．在其上的脚本主要基于 **dronekit** 和 **pyserial** 两个第三方库编写。 在与人体识别模块的通讯代码中,通过**设置超时**和编写**消息队列**以达到对人体识别模块返回数据的正确判断和应用。 而在与飞控相关的代码中,主要通过 **dronekit** 库中已有的库函数对无人

机进行控制. 同时该模块还被用于控制抛投器, 通过对树莓派引脚进行设置让其输出一定占空比的 **PWM** 波,从而达到控制抛投器的功能。在该模块设计时,考虑到作品需要有良好的时效性,于是采用了**三线程**分别运行以上三项功能. 三个线程通过预先声明的标志位变量进行交互,具有良好的时效性及效率, 同时也增加了作品的鲁棒性.

(3). 人体识别模块

人体识别模块的核心是利用 RT-Thread 团队开发的 AI 套件：**RT-AK** 将我们所训练的神经网络模型一键部署到 RT-thread 项目中，通过编写应用层代码，进行人体识别。

同时，神经网络部署至 RT-thread 项目也存在以下几点限制：

1. 神经网络每一层卷积输出的大小不宜过大，过大会增加计算复杂程度，对开发板算力要求更高；
2. 模型的分支结构数量应当较少，分支过多，内存的可复用性过低，需要内存就越大；
3. 算子支持也有限，毕竟不是在电脑端进行推理，可能会存在所不支持的算子。

考虑到以上的这些限制，我们认为关键的问题就在于神经网络的大小和复杂程度，要让神经网络在 Art-Pi 上成功运行，我们必须着眼于得到简单并且准确率高的网络。最初我们通过自己搭建小型的适用于微处理器的模型，实现简单的二分类(人体和非人体)，然后再逐渐增加网络层数提高准确率。最终虽然实现了识别的效果，但是延迟很高且准确率较低，考虑到我们的应用场景，并不适用。我们通过查找资料(如表 2 所示)，了解到了 **yolo-fastest** 算法——非常轻量级的目标检测网络，它的参数量仅有 yolo-v5 的 1/30，推理时间也仅有其 1/80。然而 yolo-fastest 对于 Art-Pi 算力来说也过于复杂，因此我们需要对其进行改进，我们进行了简单改进：

- **删减神经网络层数**，由 109 层删减为 26 层
- **去掉特征金字塔输出**，只保留一个输出，确保对人体的检测有效。

最终在经过模型转换和量化所得到的 tflite 权重文件仅有 144k，如表 1 所示，成功部署在 RT-Thread 项目后，每张图片推理时间约 50ms。

Model	MACC	Inference	Size
person_detection.tflite	6.5 M	50ms	144k

表 1 量化后的模型相关数据

Network	Model Size	mAP(VOC 2017)	FLOPS
Tiny YOLOv2	60.5MB	57.1%	6.97BFlops
Tiny YOLOv3	33.4MB	58.4%	5.52BFlops
YOLO Nano	4.0MB	69.1%	4.51Bflops
MobileNetv2-SSD-Lite	13.8MB	68.6%	&Bflops
MobileNetV2-YOLOv3	11.52MB	70.20%	2.02Bflos
Pelee-SSD	21.68MB	70.09%	2.40Bflos
Yolo Fastest	1.3MB	61.02%	0.23Bflops
Yolo Fastest-XL	3.5MB	69.43%	0.70Bflops
MobileNetv2-Yolo-Lite	8.0MB	73.26%	1.80Bflops

表 2 9 种目标检测网络

第三部分　完成情况及性能参数

(1). 续航能力：

在保持无人机处于 4m 定高状态，负重约为 400g 的情况下测得：

电池型号	电压(V)	电量（mAh）	续航时间
凤凰 3s5200mAh	11.1	5200	6 分 30 秒
格氏 4s4000mAh	14.8	4000	11 分 30 秒
格氏 4s5300mAh	14.8	5300	15 分 30 秒

表 3 电池种类与续航时间关系表

该性能参数用于分析所能携带的物资重量以及对应的负载下的续航时间基于以上实验，我们选用格式 4s 5300mAh 作为无人机的电源以获得最理想的续航能力。

(2). 飞行高度/速度：

所有实验于白天，晴朗，微风天气条件下进行。识别准确率由单次巡回过程中 10 次改变实验者位置并统计识别成功率测得。每个高度或者飞行速度测 3 次巡回。

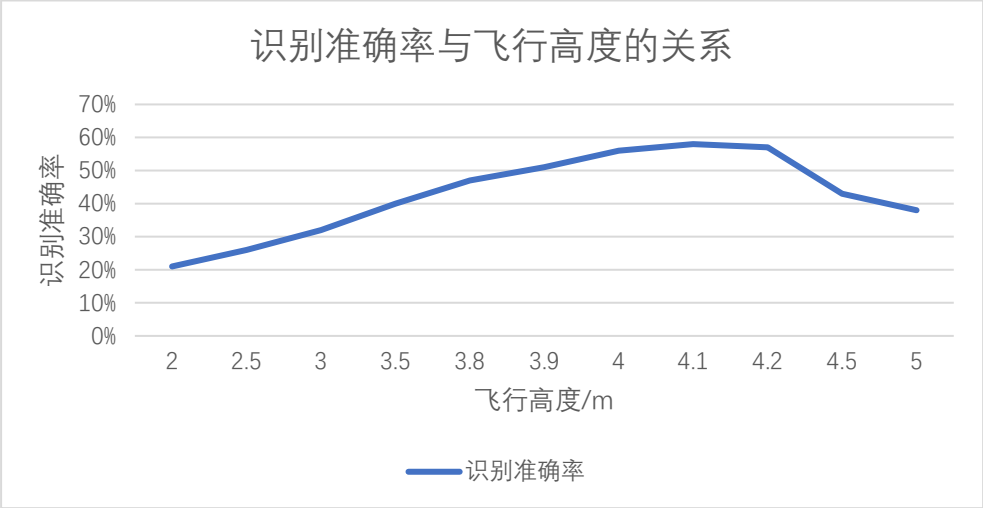


图 20 识别准确率与飞行高度折线图

分析：

识别准确率随飞行高度的上升先上升，后下降，在 4 米左右达到峰值。在 3.8 到 4.2 左右均能得到较好的识别结果。所以我们小组决定以 4 米为中心点，允许无人机在其周围浮动。高度的测定主要由加速度计测定，辅助以气压计修正以确保测得的高度以及设定的高度与实际高度差别不大。

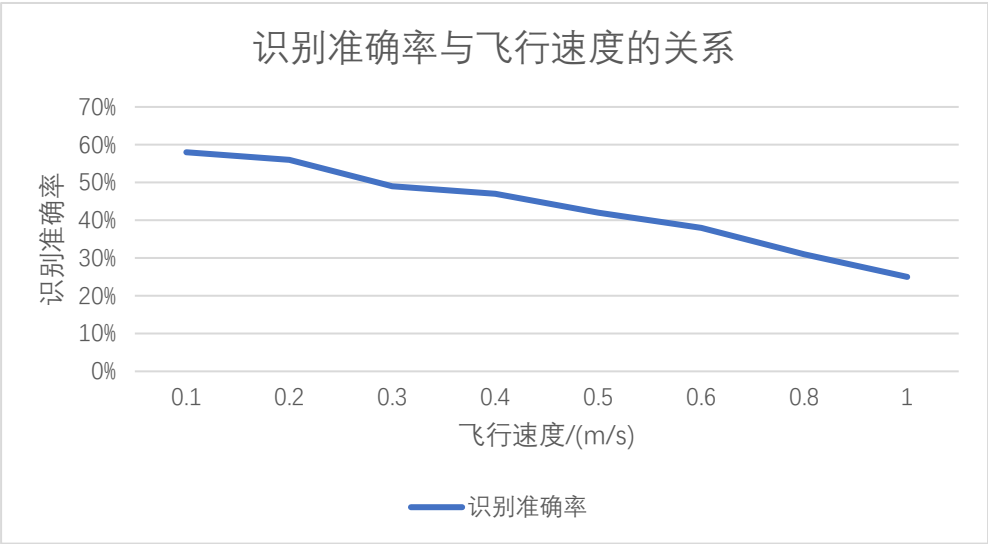


图 21 识别准确率与飞行速度折线图

分析：

识别准确率随飞行高度的上升而下降，在 0.1，0.2 米每秒左右达到峰值。但是考虑到搜寻效率以及续航时间的影响，我们小组认为在 0.2 米每秒左右达到任务最高效率。速度的测定主要由加速度计与陀螺仪测定，以确保测得的速度以及设定的速度与实际速度差别不大。

(3).巡航能力

巡航能力是指无人机按照规定路径行进的能力，在实验中，我们采用标志物的方式用以判定无人机的巡航能力，通过改变无人机的速度，观察在不同速度下，无人机能到达标志物的个数，以及返回起始点的误差。实验过程中，我们设定了 5 个目标点，且保证电池电量充足，无外界风力影响。

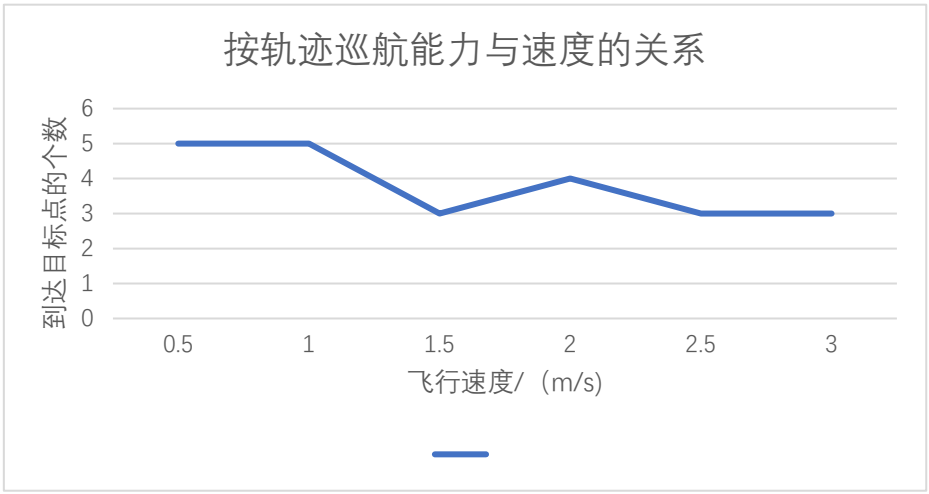


图 22 巡航能力与速度的折线图

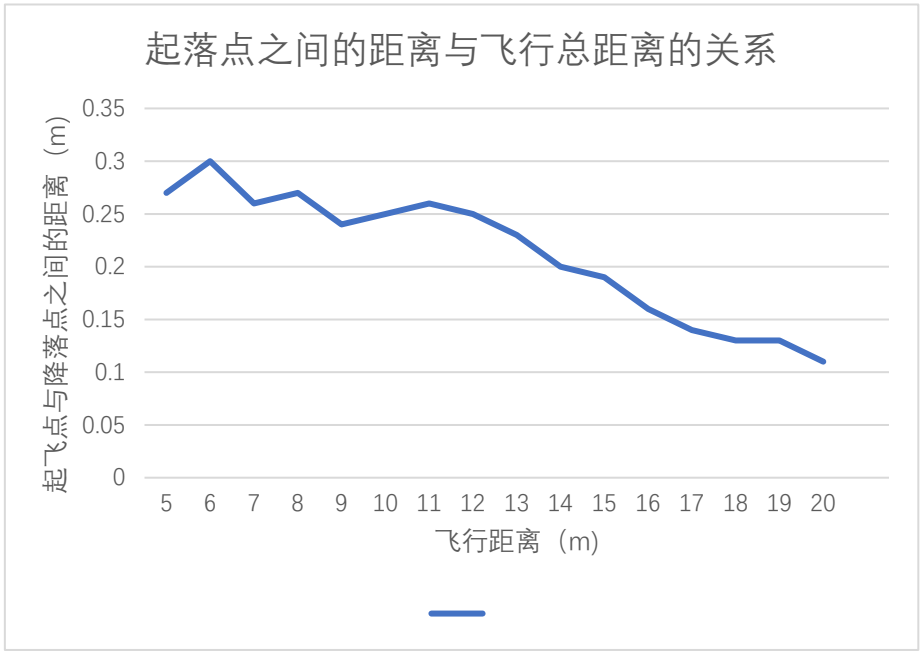


图 23 起落点距离与飞行总距离的折线图

由实验所得出的图 22 与图 23 可知，随着飞行速度的增加，到达目标点的个数总体上是呈下降趋势的，随着飞行总距离的增加起落点间的距离呈下降趋势。可见飞行速度会影响无人机上 GPS 的定位从而导致巡航能力下降，因此如果需要无人机达到良好的巡航效果，无人机速度应在 1m/s 左右，飞行总距离也可以在电量充足的情况下增加。

第四部分　总结

4.1 成果

总的来说，我们基于嵌入式 AI 实现的智能投放无人机实现了以下这些功能：

- 人体识别
- 物品投放
- 自动巡航
- 远程控制

- 飞行数据返回

通过对神经网络的删减、量化，我们成功将轻量化的 AI 模型部署在了 Art-Pi 上，实现人体识别。为无人机安装上了抛物器，通过 PWM 波控制，实现物品投放。基于 MAVLink 协议，我们实现了树莓派与飞控 pixhawk 间的通讯。PC 端可远程连接树莓派，通过在 PC 端运行程序可实现无人机的自动巡航，同时也会接收到飞控返回的无人机飞行数据。

4.2 可扩展之处

(1). 远距离数据传输:将触发投放功能时的实时场景以及位置信息返回,方便工作人员的对情况进行判断

(2). 多物体检测,不同物品投放:利用多物体检测,向不同目标投放不同物资

(3). 应用场景切换:由于利用无人机在空中进行运动限制较小,也可以用于在高速路上对违规人员(主要是车辆故障后的人员站位检测)进行检测及报警

(4). 提高识别准确率：无人机识别准确率主要受限于 Art-Pi 的算力，我们可采用算力更强的开发板来运行更复杂、准确率更高的 AI 模型：如目前的 K210 以及将要推出的 A1000。

(5). 提高负载能力：使用更大的无人机机架，更大的桨叶，更加强力的电池可有效提高无人机的负载能力。

4.3 心得体会

通过所搭载的人体识别模块，我们成功地给无人机赋予了智能，利用 Art-Pi 开发板丰富的拓展接口，实现了显示功能，便于开发过程中的调试，通过 UART 通讯协议实现了 Art-Pi 与树莓派的通讯，并且利用树莓派输出 PWM 信号来控制舵机的开合。

在尽可能地用上 Art-Pi 引脚的同时，我们还采用机载电脑树莓派来控制无人机而不是传统的遥控器控制，更能体现出无人机名字中“无人”的特点。

该无人机系统整个开发周期超过了两个月，组内同学也是在自己课后的空闲时间进行相应知识学习。整个开发的过程是不断发现问题，解决问题的过程，是不断学习的过程。我们除了相关的嵌入式 AI，无人机，串口通讯，stm32 等与作品直接相关的知识，还学了激光切割，3D 打印，电路图绘制等原件制作相关的知识。

当然开发过程肯定是充满各种困难的，比如摄像头突然不工作，开发板烧坏等等突发问题，一个小小的问题往往会折磨我们好些天。但是这些困难都只是成功路上的台阶，就要学会把他们踩在脚下。可怕的不是困难而是面对困难轻言放弃。

第五部分 参考文献

[1]裴林,李东.旋翼无人机在水上救援中的应用研究[J].黑龙江科学,2018,9(01):28-29.

[2]粟智.无人机在海上救援行动中的应用现状及发展探究[J].中国设备工程,2021(11):242-243.

[3]卢姗姗,王伟.无人机在海上救援行动中的应用现状及发展展望[J].医疗卫生装备,2019,40(02):94-98.DOI:10.19745/j.1003-8868.2019049.

第六部分　附录

1.人体识别模块代码 github 地址：https://github.com/Rich-King395/person_detection
2. Art-Pi UART 串口通讯代码 github 地址：https://github.com/Rich-King395/Serial
3. 无人机控制,无人机控制备份(可用),远端控制抛物器代码 github 仓库地址：https://github.com/Sharchxia/SmallProgram/tree/main/competition
4. Art-Pi 输出 PWM 信号项目地址 https://github.com/Rich-King395/Art-Pi-PWM.git